APPARATUS AND METHOD FOR TESTING NON-DETERMINISTIC DEVICE DATA

FIELD OF THE INVENTION

The invention relates generally to automatic test equipment, and more particularly to circuits and methods for enabling the testing of non-deterministic semiconductor device data.

BACKGROUND OF THE INVENTION

Test is an important step in the manufacture of semiconductor devices. The automatic test equipment (ATE) employed to carry out this task comprises sophisticated electronics capable of sending test signals to, and capturing output signals from, one or more devices under test (DUTs). ATE channel hardware, typically referred to as "channels", orchestrate this back and forth flow of signals.

Conventional channel circuitry, as shown in Figure 1, feeds tester data signals (drive data) originating from a pattern generator 12 to a device-under-test (DUT) 14 via interface circuitry commonly referred to as pin electronics 16. Response signals from the DUT are captured and compared to expected data with the resulting comparison data fed to a failure processor 18 in order to determine pass or fail conditions. The "expected" and "drive" data are typically programmed in the pattern generator vector memory (not shown) to occur at precise timings, in accordance with how the DUT should behave. If the data captured from the DUT fails to correspond with an expected condition, the device is considered to have failed that aspect of the test.

Modern semiconductor devices are trending towards employing multiple processing cores on the same piece of silicon, or chip. Adding to this complexity is the overall trend towards implementing asynchronous on-chip communication protocols. The end result is an exponential increase in the chip gate count, yet only modest increases in the available pin counts. Consequently, multiple sub-circuits often share the pins (interface).

This shared interface scheme is illustrated generally in Figure 2, where a plurality of device-under-test subcircuits 20a - 20c send data packets to a DUT communications port 22 . The communications port serves as the gatekeeper to accessing the DUT output pin 24. Each of the subcircuits may be clocked by a separate clock having a frequency different from not only the other subcircuits, but

also possibly different from the communications port clock. An asynchronous arbitrator 26 handles the sequencing of the data packets to the DUT output pin.

During typical DUT operation, as shown in Figures 3A and 3B, the shared interface scheme may cause problems (for conventional ATE) known as "cycle slipping", and "out-of-order data". Cycle slipping often results from the communications port clock operating at a frequency different than that of the subcircuit clocks. The result may be that the DUT output pin sees periods of "idle" data, or a number of cycles of non-packetized information. These idle periods may occur at the beginning of a data transmission, or between packets of data.

Out-of-order data often results from the subcircuits attempting to access the communications port 22 (Fig. 2) on the same clock edge, or having differing delays due to environmental conditions. Figure 3B illustrates the general concept on how an expected sequencing may be disturbed into an out-of-order packet sequence.

Both "cycle slipping" and "out of order" data present unique challenges to automatic test equipment. As previously described in the context of Figure 1, traditional ATE relies on the comparison of expected data, at expected timings, to actual data and actual timings. Providing unknown and unexpected delay and data sequences in the actual DUT data for conventional ATE often results in post-test data descrambling to determine whether the device failed or passed. This may involve substantial modifications to the test program and create substantial overhead in program development and test time.

What is desired and currently unavailable is a test solution for non-deterministic data that provides substantially real-time validation results and maximizes flexibility for the device manufacturer while reducing test costs. The apparatus and method of the present invention provides such a solution.

SUMMARY OF THE INVENTION

The present invention provides the ability for automatic test equipment to quickly validate non-deterministic data received from a device-under-test having serial or serial-like ports. This ability is available with little impact to the automatic test equipment, and is protocol independent. With the availability of such a solution, users of the automatic test equipment with experience significant test throughput improvements and reduced test costs.

To realize the foregoing advantages, the invention in one form comprises automatic test equipment for testing non-deterministic packet data from a device-under-test. The automatic test equipment includes a memory for storing expected packet data and a receiver for receiving the packet data from the device-under-test. A data validation circuit is coupled to the receiver for validating non-deterministic packet data based on the expected packet data from the vector memory.

In a further form, the invention comprises automatic test equipment for testing non-deterministic packet data from a device-under-test, the automatic test equipment includes means for storing expected packet data, and means for receiving non-deterministic packet data from the device-under-test. The equipment further includes means for validating non-deterministic packet data based on the expected packet data from a vector memory.

In yet another form, the invention comprises a method for testing non-deterministic packet data using automatic test equipment having a memory for storing expected packet data. The method includes the steps of receiving actual non-deterministic packet data from a device-under-test; and validating the non-deterministic packet data based on the expected packet data from the vector memory.

Other features and advantages of the present invention will be apparent from the following detailed description when read in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will be better understood by reference to the following more detailed description and accompanying drawings in which

FIG. 1 is a high-level block diagram of a conventional ATE architecture for driving data to a DUT and comparing the DUT response data to expected data;

FIG. 2 is a high-level block diagram of a DUT output interface architecture;

FIG. 3A is a block diagram illustrating the cycle slipping problem resulting from the DUT output scheme of Figure 2;

FIG. 3B is a block diagram illustrating the out of order problem resulting from the DUT output scheme of Figure 2;

FIG. 4 is an elevated perspective view of automatic test equipment incorporating the data validation circuit of the present invention;

FIG. 5 is a block diagram of a circuit for testing non-deterministic data according to one form of the present invention;

FIG. 6 is a block diagram of a circuit for testing non-deterministic data according to an alternative embodiment of the present invention;

Fig. 7 is a flowchart of a method according to one form of the present invention for testing non-deterministic data with the circuit of Figure 6; and

FIG. 8 is a flowchart of a method in accordance with another form of the present invention for testing non-deterministic data with the circuit of Figure 6.

DETAILED DESCRIPTION OF THE INVENTION

The present invention provides a way to test semiconductor devices that generate non-deterministic output packet data exhibiting cycle-slipping, or out-of-order results. This is accomplished through the use of automatic test equipment 30 that employs a non-deterministic data validation circuit 50 capable of analyzing the non-deterministic packet data for a real-time determination of data validity.

Referring now to Figure 4, the automatic test equipment 30, often referred to as a semiconductor tester, includes a main console 32 coupled to a testhead 34. The testhead houses the various instruments desired by the user to adequately test a device-under-test DUT 40. The instruments generally comprise large circuit boards, or channel cards, that mount within the testhead in order to interface with the DUT in a controlled fashion.

With reference now to Figure 5, the data validation circuit 50 of the present invention is preferably employed on a high-performance instrument capable of generating and receiving test signals to and from serial and serial-like data ports in accordance with SerDes and SerDes-like protocols. The circuit includes a pattern generator 52, a processing circuit 70, and pin electronics circuitry 60 for interfacing the pattern generator and the processing circuitry with the DUT 40.

Further referring to Figure 5, the pattern generator 52 includes a memory controller 54 responsive to a start signal for initiating access of a vector memory 58 to output data. In one form, the circuitry may be employed as a field-programmable-gate-array (FPGA) for high integration at low cost. Generally, test patterns stored in the vector memory include the cycle-by-cycle information for application to the DUT pins, and also the expected cycle-by-cycle data responsive to the applied test signals.

In some applications, however, the memory for pattern generation may be housed in the tester main console 32. Consequently, the location of the memory is unimportant for purposes of the present invention.

With continued reference to Figure 5, the pin electronics circuitry 60 comprises high-speed drivers 62 and receivers 64 to provide a communication interface between the tester and the DUT 40. The present invention lends itself well to flexible channel architectures such that single-ended and differential drivers and receivers may be employed. Such constructions are well-known to those skilled in the art. A preferred pin electronics channel architecture that advantageously provides a flexible AC/DC coupled environment between the tester 30 and the DUT 40 is described in co-pending U.S. Patent Application Serial Number _____, titled Hybrid

AC/DC Coupled Channel For Automatic Test Equipment, filed March 31, 2003, assigned to the assignee of the present invention, and expressly incorporated herein by reference.

With continued reference to Figure 5, the processing circuit 70 provides the tester 30 (Fig. 4) with the capability of quickly evaluating non-deterministic packet data received from the DUT 40 (Fig. 4). The circuit includes a pattern filter 72 that optionally screens or masks idle data (or timing sequence data) from the pattern generator 52 to generate filtered expect data. A first-in-first-out (FIFO) circuit 74 is disposed in the receive data path and couples to the pin electronics receiver circuit 64 to pipeline the received packet data for comparison to the filtered expect data by a match circuit 76. The match circuit preferably includes a comparator (not shown) that provides a signal back to the pattern filter 72 and the FIFO 74 to load new data following successful data matches, as more fully described below.

In operation, drive data is sent from the pattern generator 52 via the pin electronics drivers 62 to the DUT input ports. Expected data is also sent by the pattern generator to the pattern filter 72, while the actual data response signals from the DUT 40 are received by the pin electronics receivers 64. In a situation involving "cycle-slipping", the actual data may not show up at the originally expected cycle of tester operation, but yet still reflect an acceptable device signal behavior.

The processing circuit 70 handles the cycle-slipping situation by sequencing the data through the FIFO 74 in order of whatever packet arrived first, and comparing each packet of actual data to the filtered expected data (idle expected cycles filtered out). Following a successful data match, both the FIFO and the pattern filter 72 are loaded with new data to allow subsequent comparisons of packet data. If each piece of expected packet data matches each piece of actual data, then the device is deemed to have passed the test. If by the end of the test, the pattern generator 52 has not incremented through its expected data, then the DUT 40 is considered to have failed the test.

Referring now to Figure 6, a data validation circuit according to a second embodiment of the present invention, generally designated 80, provides the tester with the capability of handling packet data susceptible to both cycle-slipping and out-of-order results. The data validation circuit includes a pattern generator 82 with a memory controller 83 and a vector memory 84 (disposed locally inside the testhead or in the main tester console) that stores drive data similar to that stored in the first embodiment. However, the expected data is handled a bit differently. Instead of

storing expected packet data into the vector memory, pre-calculated "signatures" are stored. The signatures correspond to calculated checksums based on acceptable output sequences for the DUT packet data. For some protocols, the number of acceptable signatures are relatively few, allowing for this kind of pre-calculated data signature validation.

With continued reference to Figure 6, the vector memory 84 cooperates with a signature generator 86 in order to validate data received from the DUT 40 by pin electronics receivers 88. A signature generator such as a CRC arithmetic register, or linear feedback shift register is acceptable for this application. As noted above, the signature generator calculates a checksum (or "signature") from the sequence of actual data from the DUT, which is capable of being compared to the "expected" signature stored by the vector memory 84. The actual data from the DUT is also fed to a capture memory 90 to preserve the capability of analyzing the data in-depth following the test.

In operation, as shown generally in Figure 7, the user of the test equipment precalculates all known passing signatures, at step 100, corresponding to the expected data from the DUT 40. The test is then performed on the DUT, at step 102. Data captured from the DUT is then processed by the signature generator to form an actual signature, which is compared to the expected signature for a pass/fail determination, at step 104. The signature is then data logged, and the DUT binned, at step 106, according to its test results. This data signature scheme provides a predictable signature for a given stream of data, regardless of whether the data is affected by an out-of-order condition or cycle-slipping condition. Consequently, this allows the tester to evaluate a given DUT that may operate flawlessly under a wide range of operating conditions.

Alternatively, the signature method for evaluating non-deterministic data for automatic test equipment described above may employ an adaptive algorithm for those devices that may provide numerous passing signatures. This adaptive algorithm is illustrated generally in Figure 8, and includes first running a test, at step 200. A determination is then made, at step 202, as to whether the detected signature from the DUT is already stored in the vector memory 84. If yes, then the signature is data logged, and the DUT binned, at step 204. If no, then the actual test data is transferred from the capture memory 90 to a processor (not shown), at step 206, to determine whether that sequence of data passed or failed, at step 208. If the data passed, then its signature is generated, at step 210, and added to the passing signature library, at step

212. If the data failed, then the failing signature is generated, at step 210, and added to the failing signature library, at step 212.

Those skilled in the art will recognize the many benefits and advantages afforded by the present invention. Of significant importance is the throughput improvement made possible by the data validation circuit to evaluate non-deterministic data in substantially real-time. This is true for both the FIFO-based and signature generator based implementations of the data validation circuit. Further, the circuit is protocol independent, thus serving a wide variety of serial and serial-like ports for semiconductor devices.

While the invention has been particularly shown and described with reference to the preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and detail may be made therein without departing from the spirit and scope of the invention. For example, the match circuit 76 is described herein as preferably comprising a comparator, although any circuit capable of performing a matching, or comparison, function may be utilized. In a similar manner, the signature generator is described as performing a checksum function to generate a signature. This is but one way to generate a signature for non-deterministic packet data. Many other ways of generating signatures are known in the art, and are within the scope of the present invention in the context of automatic test equipment.